IN THE CLAIMS

1. (currently amended) An apparatus ~~having~~ comprising:

a KASUMI round circuit for generating a fractional portion of a KASUMI cipher; and

a calculation controller, operably coupled to ~~a calculation controller for~~ the KASUMI round circuit, sequencing the KASUMI round circuit to produce a KASUMI cipher output.

2. (previously amended) The apparatus of claim 1, further comprising a sub-key generator, connected to the calculation controller, for producing sub-keys for use by the KASUMI round circuit.

3. (currently amended) An apparatus, operable to perform ~~for performing~~ KASUMI ciphering on a KASUMI input with a key to produce a KASUMI output, comprising:

a calculation controller for generating a first control signal and a second control signal;

a KASUMI round circuit for generating a fractional portion of a KASUMI cipher, configurable in response to the first control signal for calculation of even and odd rounds to produce the KASUMI output;

memory for storing the output of the KASUMI round circuit; and

a selector for providing the input to the KASUMI round circuit in response to the second control signal, the KASUMI input being selected during the first round and a stored KASUMI round circuit output from the memory being selected during subsequent rounds.

4. (previously amended) The apparatus of claim 3, further comprising a sub-key generator for generating sub-keys for the KASUMI round circuit based on the key.

5. (original) The apparatus of claim 3, wherein the apparatus is adapted to be operable in an access point.

6. (original) The apparatus of claim 3, wherein the apparatus is adapted to be operable in an access terminal.

7. (original) The apparatus of claim 3, wherein the apparatus is adapted to be operable in a W-CDMA system.

8. (previously amended) A KASUMI round circuit for receiving an input and producing an output, operable with a partial round calculator from which the output is produced, comprising:

a memory for storing an intermediate value from the partial round calculator; and

a selector for selecting between the input and a stored intermediate value from the memory for delivery to the partial round calculator.

9. (previously amended) A KASUMI round circuit for receiving a 64-bit input and producing a 64-bit output comprising:

an FO circuit having an FO output;

an FL circuit having an FL output;

an XOR gate having a first XOR operand, a second XOR operand, and an XOR output;

a first register having a first register output;

a second register, having a second register output, the second register receiving the XOR output, the second register output being concatenated with the first register output to produce the 64-bit output;

a first input mux having a first input mux output, the first input mux selecting between an upper half of the 64-bit input and the second register output, under control of an input select signal, the first input mux output being received at the first register;

a second input mux having a second input mux output, the second input mux selecting between a lower half of the 64-bit input and the output of the first register under control of the input select signal, the second input mux output being delivered as the second XOR operand;

a first datapath mux having a first datapath mux output, the first datapath mux selecting between the first input mux output and the FO output under control of a data flow signal, the first datapath mux output delivered to the FL circuit;

a second datapath mux having a second datapath mux output, the second datapath mux selecting between the FL output and the first register output under control of the data flow signal, the second datapath mux output delivered to the FO circuit; and

a third datapath mux having a third datapath mux output, the third datapath mux for selecting between the FL output and the FO output under control of the data flow signal.

10. (previously amended) An apparatus for receiving an input and producing an output, operable with a partial FO calculator from which the output is produced, comprising:

a memory for storing an intermediate value from the partial FO calculator; and

a selector for selecting between the input and a stored intermediate value from the memory for delivery to the partial FO calculator.

11. (previously amended) An apparatus for receiving a 32-bit input and producing a 32-bit output, comprising:

a first XOR gate having a first operand, a second operand, and a first XOR output, the first operand of the first XOR gate receiving a KO sub-key;

an FI circuit having an FI input and an FI output, the FI input receiving the first XOR output;

a second XOR gate having a first operand, a second operand, and a second XOR output, the first operand of the second XOR gate receiving the FI output;

a first register having a first register output, the first register receiving the second XOR output;

a second register having a second register input and a second register output;

a first input mux having a first input mux output, the first input mux selecting between an upper half of the 32-bit input and the second register output under control of an input select signal, the first input mux output delivered to the second operand of the first XOR gate; and

a second input mux having a second input mux output, the second input mux selecting between a lower half of the 32-bit input and the first register output under control of the input select signal, the second input mux output delivered as the second operand of the second XOR gate and delivered as the second register input, the second input mux output concatenated with the second XOR output to produce the 32-bit output.

12. (previously amended) An apparatus for receiving an input and producing an output, operable with a partial FI calculator from which an intermediate value and the output is produced, comprising:

a memory for storing the intermediate value from the partial FI calculator; and

a selector for selecting between the input and a stored intermediate value from the memory for delivery to the partial FI calculator.

13. (previously amended) An apparatus for receiving a 16-bit input and producing a 16-bit output, comprising:

a first register having a first register input and a first register output;

a second register having a second register input and a second register output;

a first input mux having a first input mux output, the first input mux selecting between the first register output and an upper nine bits of the 16-bit input under control of an input select signal;

a second input mux having a second input mux output, the second input mux selecting between the second register output and a lower seven bits of the 16-bit input under control of the input select signal;

an S9 circuit, having an S9 output, receiving the first input mux output;

a first XOR having a first operand, a second operand, and a first XR output, the first operand of the first XOR receiving the S9 output and the second operand of the first XOR receiving the zero-extended second input mux output;

an S7 circuit, having an S7 output, for receiving the second input mux output;

a second XOR having a first operand, a second operand, and a second XOR output, the first operand of the second XOR receiving the truncated first XOR output and the second operand of the second XOR receiving the S7 output, the second XOR output concatenated with the first XOR output to produce the 16-bit output;

a third XOR having a first operand, a second operand, and a third XOR output, the third XOR output delivered to the first register input, the first operand of the third XOR receiving a first KI sub-key, and the second operand of the third XOR receiving the first XOR output; and

a fourth XOR having a first operand, a second operand, and a fourth XOR output, the fourth XOR output delivered to the second register input, the first operand of the fourth XOR receiving a second KI sub-key, and the second operand of the fourth XOR receiving the second XOR output.

14. (canceled)

15. (canceled)

16. (currently amended) A method for performing KASUMI ciphering on a KASUMI input, having an upper half and a lower half, to produce a KASUMI output, comprising:

generating a select signal in a calculation controller for each of eight rounds;

selecting a calculation input in accordance with the select signal, wherein the KASUMI input is selected as a the calculation input during the first round; ~~selecting~~ and a stored result is selected as the calculation input in subsequent rounds;

calculating a partial result in a KASUMI round circuit with the selected calculation input; ~~and~~

storing the partial result as the stored result in a memory; and

delivering the stored result as the KASUMI output after eight partial results are stored.

17. (previously amended) The method of claim 16, wherein the calculating step comprises:

when the round is odd:

performing an FL function on an upper half of the selected calculation input to produce an FL output;

performing an FO function on the FL output to produce an FO output; and

XORing the FO output with a lower half of the selected calculation input to produce an XOR output;

when the round is even:

performing the FO function on an upper half of the stored result to produce the FO output;

performing the FL function on the FO output to produce the FL output; and

XORing the FL output with a lower half of the stored result to produce the XOR output;

delivering as the partial result the XOR output concatenated with the upper half of the selected calculation input.

18. (original) The method of claim 16, further comprising generating sub-keys for each round.

19. (previously amended) A method for performing an FO function comprising:

for each of three stages:

selecting an input as a calculation input in the first stage;

selecting a stored result as the calculation input in subsequent stages;

calculating a partial result with the selected calculation input;

storing the partial result as the stored result in memory; and

delivering the partial result as the output.

20. (previously amended) The method of claim 19, wherein the calculating step comprises:

XORing an upper half of the selected calculation input with a sub-key to form a first XOR result;

performing an FI function on the first XOR result to form an FI result;

XORing the FI result with a lower half of the selected calculation input to form a second XOR result; and

delivering as the partial result an upper half of the selected calculation input concatenated with the second XOR result.

21. (previously amended) A method for performing an FI function comprising:
calculating a first partial result with a partial result circuit using an input;
XORing the first partial result with a sub-key to form an XORed partial result;
storing the XORed partial result in a memory as a stored result;
calculating a second partial result with the partial result circuit using the stored result; and
delivering the second partial result as the output.

22. (previously amended) The method of claim 21, wherein the partial result circuit is operable to perform the following:

performing an S9 function on an upper nine bits of the input or stored result;

zero extending a lower 7 bits of the input or stored result;

XORing the zero-extended input or stored result with the output of the S9 function to form a first XOR result;

performing the S7 function on the lower 7 bits of the input or stored result;

truncating the XOR result;

XORing the truncated first XOR result with the output of the S7 function to form a second XOR result; and

delivering as the partial result the second XOR result concatenated with the first XOR result.

23. (canceled)

24. (canceled)

25. (canceled)